A Trust-Region based Sequential Linear Programming Approach for AC Optimal Power Flow Problems

L. P. M. I. Sampath[a,*], Bhagyesh V. Patil[b], H. B. Gooi[c], J. M. Maciejowski[d], K. V. Ling[c]

[a] *Interdisciplinary Graduate School, Nanyang Technological University, Singapore*
[b] *Cambridge Centre for Advanced Research and Education in Singapore (CARES)*
[c] *School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore*
[d] *Department of Engineering, University of Cambridge, United Kingdom and Energy Research Institute at NTU (ERI@N), Singapore.*

Abstract

This study proposes a new trust-region based sequential linear programming algorithm to solve the AC optimal power flow (OPF) problem. The OPF problem is solved by linearizing the cost function, power balance and engineering constraints of the system, followed by a trust-region to control the validity of the linear model. To alleviate the problems associated with the infeasibilities of a linear approximation, a feasibility restoration phase is introduced. This phase uses the original nonlinear constraints to quickly locate a feasible point when the linear approximation is infeasible. The algorithm follows convergence criteria to satisfy the first order optimality conditions for the original OPF problem. Studies on standard IEEE systems and large-scale Polish systems show an acceptable quality of convergence to a set of best-known solutions and a substantial improvement in computational time, with linear scaling proportional to the network size.

Keywords: Nonlinear programming, optimal power flow, sequential linear programming, trust-region method.

*Corresponding author
Email address: mohashai001@e.ntu.edu.sg (L. P. M. I. Sampath)

1. Introduction

The OPF problem optimizes the total operating cost to support efficient operation of power systems while satisfying system constraints for a nominal state [1]. In practice one needs to solve a security-constrained OPF (SC-OPF) problem which takes into account the possibility of a sudden failure of a single component (generator, transmission line, transformer, etc) in the system. This is known as the N−1 security criterion [1, 2]. The OPF problem without security constraints has been extensively investigated in the literature (see, for instance [2], and references therein). This paper addresses the OPF problem for simplicity, but the benefits of our approach extend to the context of the SC-OPF problem as well. It is well-known that the OPF problem is nonlinear and nonconvex in nature, potentially having multiple equilibrium points. Hence searching for a global solution is in principle NP-hard (cf.[2, 3, 4, 5]). Electricity market clearing strategies are mainly based on nodal prices, which are the dual variables of power balance constraints of the OPF problem. This highlights the importance of the convexity and scalability features for any algorithm to use in OPF calculations [1, 6]. In addition to this, real-world OPF problems involve very large numbers of decision variables. This makes them challenging for a solution technique, both in terms of memory and computational time requirements. Consequently there is a great need for computationally efficient techniques which can handle the nonconvex AC network constraints.

In the context of OPF, solution approaches, such as linear programming (LP) [6, 7, 8], quadratic programming (QP) [9], Lagrangian relaxation [10], and interior-point (IP) methods [11] have been extensively investigated in the literature. It is worth noting that, among all these approaches, IP methods have emerged as a promising direct solution approach for OPF problems. IP methods have proven to be a viable computational alternative for the solution of large-scale OPF problems [12]. The primal-dual logarithmic barrier IP method and its predictor-corrector variant are known to be efficient for OPF solution algorithms due to their superior computational efficiency [13]. We refer to [14]

and references therein for a detailed survey of other solution approaches for the OPF problem.

Many convexification approaches for power flow constraints have been proposed to make the AC-OPF problem computationally tractable. One of the widely used techniques in the last decade is semidefinite relaxation (SDR) which can find the global optimal solution of the OPF problem for radial networks under mild operating conditions [15, 16, 17, 18, 19]. However, in the case of meshed networks, SDR possesses a relaxation gap and necessitates the use of virtual phase shifters to recover bus voltage angles [20]. This can be an expensive task in practice. Furthermore, semidefinite programs do not scale well for large-size power systems [21]. To circumvent the scalability issue associated with SDR, recently a second-order cone relaxation (SOCR) has been introduced. The SOCR enhances the computational performance, enabling the application of the technique for OPF problems in large-scale power networks [22, 23]. Based on SOCR, two different power flow formulations are considered in the literature, namely the bus injection model [21] and the branch flow model [22, 23]. Recently, the work in [24] introduced additional linear cuts in the branch flow framework to guarantee the exactness of SOCR for active distribution power networks. Similarly, an improved quadratic convex relaxation is proposed in [21] as an extension of SDR, in which voltage magnitudes are coupled with voltage angles using additional polyhedral constraints. This improves the relaxation gap in comparison to SOCR without sacrificing the computational performance. However, a significant relaxation gap still persists in many power system cases [21]. Solution approaches based on the global optimization philosophy, such as convex envelopes [25] and decomposition methods [26] have also been reported in the literature.

In the aforementioned approaches, LP methods can be an attractive candidate for OPF problems due to their inherent scalable nature. Recent works [6, 27] have used successive linear programming (SLP)[1] principles to demon-

[1]The words 'successive' or 'sequential' are used interchangeably in the context of linear

strate this fact. Specifically, [6] has shown the scalability of LP tools against the well-known IP solver IPOPT [28] as well as the nonlinear optimization solver KNITRO [29]. In [5], rectangular form of complex quantities is used to formulate the power flow model, which disregards quasi-linear relationships of active power and bus voltage angles, and the reactive power and voltage magnitudes [5]. In addition, that formulation results in noncovex voltage limit constraints which need additional slack variables in order to be linearized.

Note that SLP approaches can suffer from poor approximation of the original OPF problem due to lack of any globalization strategy. An SLP approach starting at an arbitrary point far from a solution to the original OPF may not converge to a feasible solution. In such circumstances, trust-region (TR) based methods have proven to be a viable alternative; see for instance TR-SE [30], TR-IP [31], [32] and TR-QP [33]. In TR methods, an approximation problem is solved within a small radius (called the *trust-region*). This enables a good approximation for the original OPF to be obtained at each solution step within the given trust-region.

This paper proposes a synergistic approach based on a trust-region method and SLP for the OPF problem. Our approach is very much inspired by the recently proposed successive linearization scheme of [6] and the trust-region implementation [31]. However, our work differs in the following ways:

- Unlike [6], we use the polar form of complex quantities. This assists in capturing the quasi-linear relationship between active power and bus voltage angles, and the reactive power and voltage magnitudes for the original OPF problem.

- In addition compared to [6], we propose a trust-region radius constraint to improve the validity of the linear approximations in subsequent SLP steps.

programming approximation schemes. This work prefers to use the phrase 'sequential linear programming'.

- Further, compared to [31], instead of a penalty reformulation, we propose a simple feasibility restoration phase based on the original nonlinear constraints, in order to avoid infeasibilities of intermediate linerizations.

In brief, this work uses first-order Talyor series to construct a local linear model for the original OPF problem. A trust-region constraint is designed to ensure the validity of the constructed linear model, that is, to ensure that the original nonlinear constraints are satisfied. This is then integrated in an iterative procedure to optimize bus voltage magnitudes and angles, and active and reactive power generation. This trust-region sequential linear program (TR-SLP) terminates in a finite number of iterations, returning an OPF solution satisfying the convergence criteria (see Section 3.4). The performance of TR-SLP is tested on various benchmark IEEE and Polish systems against the SLP approach in [6], NLP solvers IPOPT [28] and KNITRO [29]. The results of TR-SLP demonstrate an acceptable quality of convergence to the best-known solution for the considered benchmark systems.

The paper presents the OPF problem formulation in Section 2, followed by the algorithm of TR-SLP in Section 3. Section 4 presents the numerical results on various IEEE networks. Finally, the paper is concluded in Section 5.

2. Mathematical Formulation

In this section, we first present the network model for a general power system and formulate the AC-OPF problem. Then, a linear programming (LP) approximation of the AC-OPF problem is derived using first-order Taylor series. This linear approximation is later embedded in an iterative procedure to form the TR-SLP algorithm (see Section 3).

2.1. Network Model

We define $\mathcal{N}$ and $\mathcal{L}$ as the set of buses and the set of transmission lines of the power system respectively, where $|\mathcal{N}| = N$ and $|\mathcal{L}| = L$. Further, let $\mathcal{G}$ ($|\mathcal{G}| = G$) be the set of generators which are connected to a subset of $\mathcal{N}$. To formulate

the OPF problem, we use the polar form of the complex bus voltage $v \in \mathbb{C}^N$ and its i^{th} element $v_i = V_i e^{\mathrm{j}\delta_i}$, where V_i is the voltage magnitude and δ_i is the phase angle of the voltage phasor v_i at bus $i \in \mathcal{N}$. Complex power generation is denoted by $s^{\mathrm{G}} \in \mathbb{C}^G$ such that $s_g^{\mathrm{G}} = P_g^{\mathrm{G}} + \mathrm{j}Q_g^{\mathrm{G}}$ for generator $g \in \mathcal{G}$, where P_g^{G} and Q_g^{G} are the active and reactive power generation respectively. These two vectors (v and s^{G}) are the decision variables of the OPF problem. The parameters involved in the formulation are defined below.

The standard $\pi-$model is applied for modeling transmission lines. For the transmission line $l \in \mathcal{L}$, let $Y \in \mathbb{C}^L$ be the branch admittance vector, having components $Y_l = g_{l(i,j)} + \mathrm{j}b_{l(i,j)}$, where $g_{l(i,j)}$ and $b_{l(i,j)}$ are the series conductance and susceptance respectively. Similarly, $b_{l(i,j)}^{\mathrm{sh}} \in \mathbb{R}$ is the line charging susceptance for tranmission line l. Complex power demand is characterized by $s^{\mathrm{D}} \in \mathbb{C}^N$ such that $s_i^{\mathrm{D}} = P_i^{\mathrm{D}} + \mathrm{j}Q_i^{\mathrm{D}}$, where P_i^{D} and Q_i^{D} are the active and reactive power demand respectively at bus i.

2.2. AC-OPF Problem Formulation

The objective function of the OPF problem is generally formulated as the generation cost minimization. The constraints are formulated to satisfy the power balance at each bus, the generation capacity margins, and network constraints, *namely* power flow limits and voltage bounds.

The quadratic cost function for generator g in the system is represented below.

$$C_g = c_{2,g}\left(P_g^{\mathrm{G}}\right)^2 + c_{1,g}P_g^{\mathrm{G}} + c_{0,g}\,, \qquad \forall g \in \mathcal{G} \tag{1}$$

where $c_{2,g}$, $c_{1,g}$ and $c_{0,g}$ denote the coefficients of quadratic, linear, and constant terms of the cost function, respectively. Then the complete OPF can be formulated as a NLP problem to optimize the total operating cost of the system:

$$\min_{\substack{\delta_i,\ V_i,\\ P_g^{\mathrm{G}},\ Q_g^{\mathrm{G}}}} \quad \sum_{g\in\mathcal{G}} C_g \tag{2a}$$

$$\text{s.t.} \quad \sum_{g\in\mathcal{G}(i)} P_g^{\mathrm{G}} - V_i^2 \sum_{j\in\mathcal{N}(i)} g_{l(i,j)}$$

$$+ V_i \sum_{j \in \mathcal{N}(i)} V_j \left[g_{l(i,j)} \cos(\delta_{i,j}) - b_{l(i,j)} \sin(\delta_{i,j}) \right] = P_i^{\mathrm{D}}, \ \forall i \in \mathcal{N} \quad (2b)$$

$$\sum_{g \in \mathcal{G}(i)} Q_g^{\mathrm{G}} - V_i^2 \sum_{j \in \mathcal{N}(i)} (b_{l(i,j)} + b_{l(i,j)}^{\mathrm{sh}}/2)$$

$$+ V_i \sum_{j \in \mathcal{N}(i)} V_j \left[b_{l(i,j)} \cos(\delta_{i,j}) + g_{l(i,j)} \sin(\delta_{i,j}) \right] = Q_i^{\mathrm{D}}, \ \forall i \in \mathcal{N} \quad (2c)$$

$$I_{l(i,j)}^2 \leq (I_l^{\max})^2, \ i,j \in \mathcal{N}, \ \forall l \in \mathcal{L} \quad (2d)$$

$$I_{l(j,i)}^2 \leq (I_l^{\max})^2, \ i,j \in \mathcal{N}, \ \forall l \in \mathcal{L} \quad (2e)$$

$$I_{l(i,j)}^2 = I_{l(i,j)}^A V_i^2 + I_{l(i,j)}^B V_j^2 - 2 V_i V_j \left[I_{l(i,j)}^C \cos(\delta_{i,j}) - I_{l(i,j)}^D \sin(\delta_{i,j}) \right] \quad (2f)$$

$$V_i \in \left[V_i^{\min}, V_i^{\max} \right], \ \forall i \in \mathcal{N} \quad (2g)$$

$$P_g^{\mathrm{G}} \in \left[P_g^{\mathrm{G,min}}, P_g^{\mathrm{G,max}} \right], \ \forall g \in \mathcal{G} \quad (2h)$$

$$Q_g^{\mathrm{G}} \in \left[Q_g^{\mathrm{G,min}}, Q_g^{\mathrm{G,max}} \right], \ \forall g \in \mathcal{G} \quad (2i)$$

where $\delta_{i,j} = \delta_i - \delta_j$; constraints (2b) and (2c) represent the active and reactive power balance at each bus; $\mathcal{G}(i)$ and $\mathcal{N}(i)$ are the set of generators connected at bus i, and the set of buses connected to bus i by transmission lines, respectively; and constraints (2d) and (2e) constrain the maximum current flow through each transmission line. Here, (2f) models the apparent current flow from bus i to bus j through transmission line l, where

$$\begin{aligned}
I_{l(i,j)}^A &= g_{l(i,j)}^2 + \left(b_{l(i,j)} + b_{l(i,j)}^{\mathrm{sh}}/2 \right)^2, \\
I_{l(i,j)}^B &= g_{l(i,j)}^2 + b_{l(i,j)}^2, \\
I_{l(i,j)}^C &= g_{l(i,j)}^2 + b_{l(i,j)} \left(b_{l(i,j)} + b_{l(i,j)}^{\mathrm{sh}}/2 \right) \text{ and} \\
I_{l(i,j)}^D &= b_{l(i,j)} b_{l(i,j)}^{\mathrm{sh}}/2 \ ;
\end{aligned}$$

The physical laws of power flow have been considered in modeling these constraints. Constraint (2g) bounds the engineering limits of the voltage at each bus; and (2h) and (2i) bound the active and reactive power generation capabilities of each generator respectively; and $(\cdot)^{\min}$ and $(\cdot)^{\max}$ indicate the lower and upper bound of the decision variables, respectively. The optimization problem

consists of $2(N+G)$ number of variables to optimize subject to the variable bounds and $2(N+L)$ number of constraints.

2.3. LP Formulation

The nonlinearity in the aforementioned OPF problem comes from equations (1), (2b), (2c) and (2f). In our proposed iterative procedure (TR-SLP), the nonlinear terms in these equations are linearized by applying first-order Taylor series approximations evaluated at the solution of the previous iteration. Assume the decision variable vector pertaining to the NLP problem (2) as

$$x\colon = \left[\delta_1, \ldots, \delta_N, V_1, \ldots, V_N, P_1^{\mathrm{G}}, \ldots, P_G^{\mathrm{G}}, Q_1^{\mathrm{G}}, \ldots, Q_G^{\mathrm{G}}\right]^{\mathrm{T}} \in \mathbb{R}^{2(N+G)}.$$

where $(\cdot)^{\mathrm{T}}$ is the transpose operator. Then, the partial derivatives of (1), (2b), (2c) and (2f) are used to compute the Jacobian matrices as follows.

$$J^{\mathrm{C},k-1} = \left[\mathbf{0}_{2N}^{\mathrm{T}},\ \frac{\partial C_1}{\partial P_1^{\mathrm{G}}}, \ldots, \frac{\partial C_G}{\partial P_G^{\mathrm{G}}},\ \mathbf{0}_G^{\mathrm{T}}\right]\Big|_{x^{k-1}} \tag{3a}$$

$$P_i^{\mathrm{N}} = V_i^2 \sum_{j \in \mathcal{N}(i)} g_{l(i,j)} - V_i \sum_{j \in \mathcal{N}(i)} V_j \left[g_{l(i,j)}\cos(\delta_{i,j}) - b_{l(i,j)}\sin(\delta_{i,j})\right],\ \forall i \in \mathcal{N} \tag{3b}$$

$$J_i^{\mathrm{P},k-1} = \left[\frac{\partial P_i^{\mathrm{N}}}{\partial \delta_1}, \ldots, \frac{\partial P_i^{\mathrm{N}}}{\partial \delta_N},\ \frac{\partial P_i^{\mathrm{N}}}{\partial V_1}, \ldots, \frac{\partial P_i^{\mathrm{N}}}{\partial V_N},\ -e_{G,i}^{\mathrm{T}},\ \mathbf{0}_G^{\mathrm{T}}\right]\Big|_{x^{k-1}},\ \forall i \in \mathcal{N} \tag{3c}$$

$$Q_i^{\mathrm{N}} = V_i^2 \sum_{j \in \mathcal{N}(i)} (b_{l(i,j)} + b_{l(i,j)}^{\mathrm{sh}}/2) - V_i \sum_{j \in \mathcal{N}(i)} V_j \left[b_{l(i,j)}\cos(\delta_{i,j}) + g_{l(i,j)}\sin(\delta_{i,j})\right],\ \forall i \in \mathcal{N} \tag{3d}$$

$$J_i^{\mathrm{Q},k-1} = \left[\frac{\partial Q_i^{\mathrm{N}}}{\partial \delta_1}, \ldots, \frac{\partial Q_i^{\mathrm{N}}}{\partial \delta_N},\ \frac{\partial Q_i^{\mathrm{N}}}{\partial V_1}, \ldots, \frac{\partial Q_i^{\mathrm{N}}}{\partial V_N},\ \mathbf{0}_G^{\mathrm{T}},\ -e_{G,i}^{\mathrm{T}}\right]\Big|_{x^{k-1}},\ \forall i \in \mathcal{N} \tag{3e}$$

$$J_{l(i,j)}^{\mathrm{I},k-1} = \left[\frac{\partial I_{l(i,j)}^2}{\partial \delta_1}, \ldots, \frac{\partial I_{l(i,j)}^2}{\partial \delta_N},\ \frac{\partial I_{l(i,j)}^2}{\partial V_1}, \ldots, \frac{\partial I_{l(i,j)}^2}{\partial V_N},\ \mathbf{0}_{2G}^{\mathrm{T}}\right]\Big|_{x^{k-1}}, \quad i,j \in \mathcal{N},\ \forall l \in \mathcal{L} \tag{3f}$$

where $\mathbf{0}_{(\cdot)} = \{0\}^{(\cdot)}$ and $e_{G,i} \in \{0,1\}^G$, in which the g^{th} element is 1 if generator $g \in \mathcal{G}(i)$, or is 0 otherwise. P_i^{N} and Q_i^{N} denote the sum of active and

reactive power extractions from bus i respectively; $(\cdot)^{k-1}$ denote the value of the decision variable/vector $(\cdot)$ at the $(k-1)^{\text{th}}$ iteration. Equations (3a), (3c), (3e) and (3f) represent the Jacobian matrices of (1), (2b), (2c) and (2f) respectively, which are originally nonlinear. At the k^{th} iteration of TR-SLP, those Jacobian matrices in (3) are updated based on the solution of the previous $(k-1)^{\text{th}}$ iteration. Finally, the LP approximation of the OPF problem (2) to be solved at the k^{th} iteration, obtained based on the solution of the $(k-1)^{\text{th}}$ iteration, can be deduced as follows.

$$\text{LP}\big(x^{k-1}\big)\begin{cases} \min\limits_{x} J^{\text{C},k-1}(x-x^{k-1}) + \sum\limits_{g\in\mathcal{G}} C_g|_{x^{k-1}} \\ \text{s.t. } J_i^{\text{P},k-1}(x-x^{k-1}) + P_i^{\text{N}}\big|_{x^{k-1}} - \sum\limits_{g\in\mathcal{G}(i)} P_g^{\text{G},k-1} = -P_i^{\text{D}},\ \forall i\in\mathcal{N} \\ \qquad J_i^{\text{Q},k-1}\left(x-x^{k-1}\right) + Q_i^{\text{N}}\big|_{x^{k-1}} - \sum\limits_{g\in\mathcal{G}(i)} Q_g^{\text{G},k-1} = -Q_i^{\text{D}}, \forall i\in\mathcal{N} \\ \qquad J_{l(i,j)}^{\text{I},k-1}(x-x^{k-1}) + I_{l(i,j)}^2\Big|_{x^{k-1}} \le (I_l^{\max})^2,\ i,j\in\mathcal{N},\ \forall l\in\mathcal{L} \\ \qquad J_{l(j,i)}^{\text{I},k-1}(x-x^{k-1}) + I_{l(j,i)}^2\Big|_{x^{k-1}} \le (I_l^{\max})^2,\ i,j\in\mathcal{N},\ \forall l\in\mathcal{L} \\ \qquad (2\text{g})-(2\text{i}) \end{cases} \tag{4}$$

It should be noted that (4) is tightly-coupled to the original OPF problem (2) at the evaluated point x^{k-1}.

3. Trust-Region based Sequential Linear Programming Algorithm

This section first introduces components such as trust-region LP formulation, feasibility restoration phase and step acceptance/rejection criterion. Then the pseudo-code of the main algorithm TR-SLP comprising all these components is presented. For ease of explanation, the AC-OPF problem (2) is represented

using a generic NLP form as follows:

$$\text{NLP} \begin{cases} \min\limits_{x} & f(x) \\ \text{s.t.} & h(x) = 0 \\ & c(x) \leq 0 \\ & x^{\min} \leq x \leq x^{\max} \end{cases} \tag{5}$$

where f represents the objective function (2a); h represents the set of equality constraints which include (2b) and (2c); c represents the set of inequality constraints which include (2d) and (2e); and $x^{\min}$ and $x^{\max}$ in (5) represent the variable bounds (2g)-(2i).

3.1. Trust-Region Linear Program

At the k^{th} iteration, the $\text{LP}\left(x^{k-1}\right)$ approximates the original OPF problem (2) at x^{k-1}. However, it may be a very poor representation of (2) if $\left\|x^k - x^{k-1}\right\|$ is not sufficiently small. To circumvent this issue, we consider bounding x^{k-1} variations within a small closed region called the trust-region Δ^k. Specifically, we add a trust-region radius constraint to the LP approximation (4) and form the following optimization problem.

$$\text{TR-LP}\left(x^{k-1}, \Delta^k\right) \begin{cases} \min\limits_{d} & f(x^{k-1}) + \left[\nabla f(x^{k-1})\right]^{\mathrm{T}} d \\ \text{s.t.} & h(x^{k-1}) + \left[\nabla h(x^{k-1})\right]^{\mathrm{T}} d = 0 \ : \lambda_h^k \\ & c(x^{k-1}) + \left[\nabla c(x^{k-1})\right]^{\mathrm{T}} d \leq 0 \ : \lambda_c^k \\ & \max(x^{\min} - x^{k-1}, -\Delta^k) \leq d \\ & d \leq \min(x^{\max} - x^{k-1}, \Delta^k) \end{cases} \tag{6}$$

where the decision variable vector $d := x - x^{k-1}$ and $\Delta^k > 0 \in \mathbb{R}^{2(N+G)}$ is the TR radius. Here, $\nabla f(x^{k-1})$, $\nabla h(x^{k-1})$ and $\nabla c(x^{k-1})$ represent the first-order partial derivatives of $f(x)$, $h(x)$ and $c(x)$ with respect to x, evaluated at x^{k-1} as in (4), respectively; λ_h^k and λ_c^k are the Lagrange multipliers of the equality (h) and inequality (c) constraints, respectively, with $\lambda^k = \left[(\lambda_h^k)^{\mathrm{T}} \ \ (\lambda_c^k)^{\mathrm{T}}\right]^{\mathrm{T}}$. The solution d^k of the above optimization problem is used as a step to define the

new solution approximation, *i.e.* $x^k = x^{k-1} + d^k$ (see Section 3.3). The Karush-Kuhn-Tucker (KKT) conditions for (6) are:

$$h(x^{k-1}) + \left[\nabla h(x^{k-1})\right]^{\mathrm{T}} d^k = 0 \tag{7a}$$

$$c(x^{k-1}) + \left[\nabla c(x^{k-1})\right]^{\mathrm{T}} d^k \leq 0 \tag{7b}$$

$$\nabla f(x^{k-1}) + \nabla c(x^{k-1})\lambda_c^k + \nabla h(x^{k-1})\lambda_h^k = 0 \tag{7c}$$

$$\left(c(x^{k-1}) + \left[\nabla c(x^{k-1})\right]^{\mathrm{T}} d^k\right) \lambda_c^k = 0 \tag{7d}$$

$$\lambda_c^k \geq 0 \tag{7e}$$

Equations (7) will be satisfied at every successful TR-LP computation.

It should be noted that a smaller TR radius may cause constraint infeasibilities or may reduce the speed of convergence. Similarly, a larger TR radius will weaken the validity of linear models that represent nonlinear constraints in (2). Therefore Δ^k is modified at each step of the algorithm (step 6 of Algorithm 1), the modification depending on the improvement in optimality.

3.2. Feasibility Restoration

In practice, TR-LP$\left(x^{k-1}, \Delta^k\right)$ can be infeasible due to the following two reasons: *i*) The constraint gradients $\left[\nabla h(x^{k-1})\right]^{\mathrm{T}}$ can become degenerate at the point x^{k-1}, leading to infeasible linearized constraints. Then the system $\left[\nabla h(x^{k-1})\right]^{\mathrm{T}} d = -h(x^{k-1})$ simply has no solution. *ii*) If the trust-region is too small, the TR-LP may be infeasible. In such circumstances, the linear constraint, $h(x^{k-1}) + \left[\nabla h(x^{k-1})\right]^{\mathrm{T}} d = 0$, cannot be satisfied within the trust-region radius Δ^k of x^{k-1}.

Feasibility restoration (NLP-FR) searches for a feasible point by solving the following problem, so that the next TR-LP subproblem to be solved will be

feasible.

$$\text{NLP-FR} \begin{cases} \min\limits_{\substack{x,\ s_c,\\ s_h^+,\ s_h^-}} & s_c + s_h^+ + s_h^- \\ \text{s.t.} & c(x) - s_c \leq 0 \\ & h(x) - s_h^+ + s_h^- = 0 \\ & x^{\min} \leq x \leq x^{\max} \\ & s_c,\ s_h^+,\ s_h^- \geq 0 \end{cases} \tag{8}$$

where $s_c,\ s_h^+,\ s_h^-$ are slack variables used to relax the inequality and equality constraints respectively.

If the NLP-FR cannot find a solution with zero objective value, then the OPF problem (2) is declared as infeasible. Otherwise, we have found a feasible point x^k, which is used to compute the step-size $d^k := x^k - x^{k-1}$.

3.3. Step Acceptance/Rejection Criterion

To accept or reject the new step-size d^k and update the trust-region radius Δ^k for the next TR-SLP iteration, we compute the ratio ρ^k between predicted and actual reduction in the cost function (2a).

Let d^k be a solution of TR-LP$\left(x^{k-1}, \Delta^k\right)$. Then the predicted reduction in the objective is

$$\Delta\phi^k_{\text{pre}} = \left[\nabla f(x^{k-1})\right]^{\mathrm{T}} d^k. \tag{9}$$

In order to take into account any constraint violations, as well as the actual value of the objective of the NLP (5), the following merit function is defined:

$$\phi(x^k) = f(x^k) + (\nu_h^k)^{\mathrm{T}} |h(x^k)| + (\nu_c^k)^{\mathrm{T}} \max\{c(x^k), 0\}, \tag{10}$$

where $\nu_h^k \in \mathbb{R}_+^{n_h}$ and $\nu_c^k \in \mathbb{R}_+^{n_c}$ are penalty factors for equality and inequality constraints respectively. These are derived in each iteration k based on (11a) and (11b) using dual variables λ_h^k and λ_c^k as follows.

$$\nu_h^k = \max\{\nu_h^{k-1},\ \lambda_h^k\} \tag{11a}$$

$$\nu_c^k = \max\{\nu_c^{k-1},\ \lambda_c^k\} \tag{11b}$$

$$\nu^0_{h,m} = \frac{\| \nabla f(x^0) \|_2}{\| \nabla h_m(x^0) \|_2}, \ \forall m \in \{1, \ldots, n_h\} \tag{11c}$$

$$\nu^0_{c,m} = \frac{\| \nabla f(x^0) \|_2}{\| \nabla c_m(x^0) \|_2}, \ \forall m \in \{1, \ldots, n_c\} \tag{11d}$$

Further, (11c) and (11d) are used to calculate the penalty factors for the first iteration, where h_m and c_m are the m^{th} equality and inequality constraints respectively; n_h and n_c are the number of equality and inequality constraints respectively. The actual reduction in the objective is

$$\Delta\phi^k_{\text{act}} = \phi(x^k) - \phi(x^{k-1}). \tag{12}$$

The ratio ρ^k is then defined as

$$\rho^k = \frac{\Delta\phi^k_{\text{act}}}{\Delta\phi^k_{\text{pre}}} \tag{13}$$

Then,

$$\Delta^{k+1} = \begin{cases} \alpha_1 \Delta^k & \text{if} \quad \rho^k \leq 0 \\ \alpha_2 \Delta^k & \text{if} \quad 0 < \rho^k \leq 0.25 \\ \Delta^k & \text{if} \quad 0.25 < \rho^k \leq 0.75 \\ \min(2\Delta^k, \Delta^{\max}) & \text{if} \quad 0.75 < \rho^k \end{cases} \tag{14}$$

where $\alpha_1 \in (0, 1)$, $\alpha_2 \in (0, 1)$ and $\Delta^{\max}$ are constants. This is a heuristic, and values for these parameters should be determined on a case-by-case basis in the context of the OPF problem (2).

Remark 1*: If $\rho^k < 0$, then the iteration is considered as a failure. In such a case, the new point x^k is rejected, and the TR radius Δ^k for the next iteration is reduced to α_1 times its present value, and the TR-LP$\left(x^{k-1}, \Delta^k\right)$ is solved again. If $\rho^k \geq 0$, then the new point $x^k = x^{k-1} + d^k$ is accepted and the algorithm proceeds to the next step with the updated TR radius Δ^{k+1}.*

We now summarize the TR-SLP algorithm via the pseudo-code in Algorithm 1, utilizing the aforementioned trust-region linear program (TR-LP), feasibility restoration (NLP-FR), and step acceptance/rejection (StepQuality) ingredients. It may be noted that the output $\left(f^k, x^k\right)$ of the TR-SLP algorithm which satisfies the *convergence* conditions given in Section 3.4, is a local optimal solution of the OPF problem (2).

Algorithm 1:

Trust-Region Sequential Linear Program (TR-SLP)

Input : f, h, c, x^0, α_1, α_2, Δ^1, $\Delta^{\max}$, K

Output: f^k, x^k

while *convergence not satisfied* **do**

  /* solve trust-region linear program */

  $(f^k, d^k, \lambda^k) \leftarrow$ solve TR-LP$\left(x^{k-1}, \Delta^k\right)$

  if TR-LP$\left(x^{k-1}, \Delta^k\right)$ *is not feasible* **then**

    /* feasibility restoration phase */

    $d^k \leftarrow$ solve NLP-FR

  end

  /* step quality determination phase */

  $\left(\Delta^{k+1}, \rho^k\right) \leftarrow$ StepQuality$\left(f, \nabla f, x^{k-1}, d^k, \alpha_1, \alpha_2, \Delta^{\max}\right)$

  if $\rho^k < 0$ **then**

    /* reject step */

    $x^k \leftarrow x^{k-1}$

  else

    /* accept step */

    $x^k \leftarrow x^{k-1} + d^k$

    $k = k + 1$

  end

end

3.4. Discussion on Convergence

We first give the necessary Karush-Kuhn-Tucker (KKT) conditions adopted from [34] for TR-SLP described in Algorithm 1. The TR-SLP stops when the following conditions are satisfied.

(a) $\| d^k \|_\infty \leq \epsilon_d$

(b) $\| h(x^k) \|_\infty \leq \epsilon$, $\max\{c(x^k)\} \leq \epsilon$

(c) $\max\{\| \nabla f(x^k) + \nabla c(x^k)\lambda_c^k + \nabla h(x^k)\lambda_h^k \|_\infty, \| c(x^k)\lambda_c^k \|_\infty\} < \epsilon_\lambda \left(1 + \| \lambda^k \|_2\right)$

(d) $\lambda_c^k \geq 0$

where ϵ_d, ϵ, and ϵ_λ are tolerances chosen for the step change, constraint satisfaction and KKT condition satisfaction respectively. Condition (a) implies that the step-size has reached the user-specified accuracy ϵ_d. Condition (b) implies that within the trust-region radius, the original nonlinear constraints h and c in the NLP (5) are satisfied to a user-specified accuracy of ϵ. Conditions (c) and (d) provide a measure of the closeness of the computed solution to a point satisfying the first-order optimality conditions for the NLP problem (5).

To establish the convergence of TR-SLP (Algorithm 1) consider the following. TR-SLP solves a trust-region linear approximation (*i.e.* TR-LP (6)) of the original NLP (5) at each iteration k. If TR-LP is feasible, it will compute d^k (step 2). Subsequently, based on the previous iterates x^{k-1} and d^k, the actual-to-predicted cost ratio ρ^k and the new trust-region radius Δ^{k+1} are determined (step 6). Then, based on ρ^k, as the TR-SLP progresses, Δ^{k+1} shrinks, ensuring the tightness of the linear approximation TR-LP (cf. Figure 4). This leads the successive iterates x^k to converge to a local solution of NLP (5), satisfying the conditions (a)–(d).

4. Numerical Experiments and Discussion

In this section, we report numerical results with the proposed TR-SLP algorithm for OPF problem (2). The TR-SLP is analyzed on a benchmark test

suite consisting of IEEE $(14, 30, 57, 118$ and $300)$ systems, and Polish 2383wp, 2746wop and 3012wp systems (the number refers to the number of buses in the respective test case) available in [35]. We note that [6] is a recent computational study on the AC-OPF problem for the aforementioned systems. It reports the OPF (2) results using general purpose optimization solvers like IPOPT [28] and KNITRO [29], and a penalty reformulations based successive linear programming (SLP) method. In [6], two types of OPF problems are solved for each system, *viz.* without line flow limits (baseline case), and with line flow limits (thermally constrained case). We obtain the line flow limits data for this work from [6, Table II] . Further, [6, Tables IV, V] report the results obtained using their proposed SLP, NLP solvers IPOPT and KNITRO for the various benchmark IEEE and Polish systems. These results have been obtained with constraint satisfaction up to 0.001 tolerance. We shall use these results in our study to compare the performance of our TR-SLP in terms of optimality and computational time.

Convergence and optimality of the solution of SLP algorithms depend on the selected initial point (cf. [6]). In this study, we consider two different initialization strategies for TR-SLP, *viz*, flat start and DC warm start, to demonstrate the variation of performance with respect to the starting point. In the flat start, we assume unit voltage phasors and half-max outputs for all generation. The DC warm start is constructed with the solution obtained from the DC-OPF problem combined with unit voltage magnitudes and half-max reactive power generation. We conduct three case studies to showcase the performance of the TR-SLP. Firstly, the computational time of the TR-SLP with the aforementioned test cases is compared against that of KNITRO, IPOPT and SLP. Secondly, we study the optimality of the solution obtained using the TR-SLP for the same test cases against the KNITRO solver run in a multi-start mode (henceforth referred as KNITRO-MS). We note that KNITRO run in a multi-start mode results in improved local optimal solutions [29]. Finally, we tighten the tolerances (*i.e.* $\epsilon_d, \epsilon, \epsilon_\lambda$ in TR-SLP) and study the relative improvement in the optimality of the solution obtained using the TR-SLP for the same test

cases, and note the trade-off against computation time.

The TR-SLP algorithm is implemented in MATLAB and the optimization problems are formulated based on the MATPOWER library [35]. The LP subproblems are solved using CPLEX 12.6 [36] and feasibility restoration subproblems using the MATLAB *fmincon* solver based on the *interior-point method*. All experiments are carried out on a desktop PC with an Intel®Core i7-5500U 4 core CPU processor running at 2.40GHz with 8GB RAM. Based on our experience with the numerical experiments reported in this work, the parameters of TR-SLP are chosen as follows: $\alpha_1 = 0.1$, $\alpha_2 = 0.25$, $\Delta^{(0)} = 0.4$, $\Delta^{\max} = 2$, $K = 30$, $\epsilon_d = 0.1$, $\epsilon = \epsilon_\lambda = 0.01$. Note that, in all our numerical studies, the original nonlinear constraints (2b)−(2e) are satisfied up to the same accuracy employed by [6] (*i.e.* 0.001 or lesser). This is specifically demonstrated for two large Polish $(2746, 3012)$ systems in columns A and B of Table 4.

4.1. Case Study 1: Computational Time Comparison

In this study, TR-SLP is executed with a flat start strategy for OPF problem (2) and is compared in Table 1 with different solution approaches for several test cases. It should be noted that in [6], four different initialization strategies, *viz.* flat start, DC warm start, AC warm start and uniform cold start are used and the best solver time recorded for each test case is reported (see Table 1; KNITRO, IPOPT, and SLP).

It can be observed that for all IEEE systems, the CPU times of KNITRO and IPOPT are almost the same. Comparatively in Polish systems, KNITRO is noted to be slower. SLP is found to be the slowest for all IEEE systems. However, its performance is observed to improve for Polish systems with respect to KNITRO and IPOPT. We observed TR-SLP to be fastest among all solution approaches in many test cases (except IEEE 300 and Polish 2383 thermally constrained cases, where KNITRO and IPOPT are slightly faster). The TR-SLP reports comparatively the best CPU time for the largest Polish 3012 system, approximately 6, 2 and 3 times faster than KNITRO, IPOPT, and SLP, respectively, for the baseline case and approximately 5, 1.5 and 2 times faster

Table 1: The Computational Time Comparison (in Seconds) among All Solvers/Algorithms. TR-SLP is Executed under a Flat Start Strategy.

Test Case	Computational Time (s)							
	Baseline Case				Thermally Constrained Case			
	KNITRO	IPOPT	SLP	TR-SLP	KNITRO	IPOPT	SLP	TR-SLP
IEEE 14	0.12	0.16	0.24	0.06	0.12	0.19	0.19	0.1
IEEE 30	0.19	0.2	0.84	0.07	0.19	0.19	0.58	0.08
IEEE 57	0.32	0.38	0.76	0.14	0.32	0.35	0.79	0.17
IEEE 118	0.84	1.18	2.53	0.94	1.23	1.18	2.86	1.01
IEEE 300	2.27	2.63	4.88	2.17	1.96	2.22	5.23	3.61
Polish 2383	38.65	88.47	37.05	15.94	43.63	26.32	36.42	28.09
Polish 2746	42.43	20.67	51.71	8.97	82.02	39.66	43.78	12.85
Polish 3012	96.51	33.15	49.73	16.40	99.53	35.56	46.34	20.07

than KNITRO, IPOPT, and SLP, respectively, for the thermally constrained case.

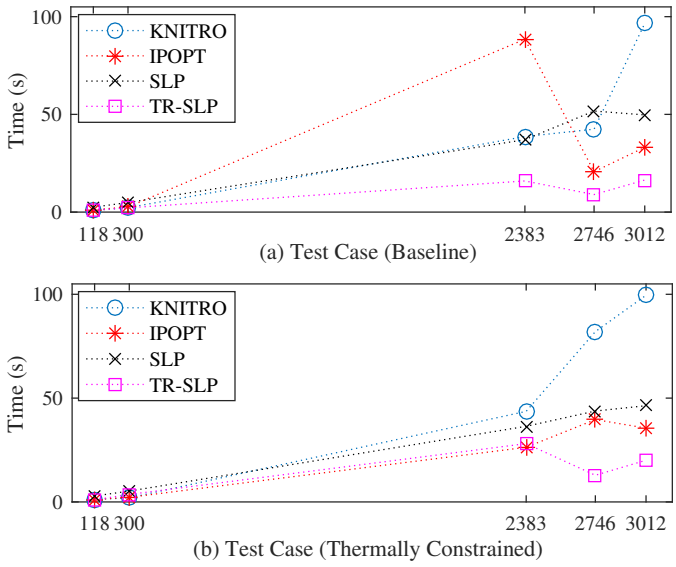


Figure 1: The computational scalability comparison among all solvers/algorithms for various IEEE (118, 300) and Polish (2383, 2746, 3012) systems. TR-SLP is simulated under the flat start strategy.

Figure 1 reports the computational time growth among all solution approaches for both the baseline and thermally constrained cases. It can be seen that both SLP and TR-SLP give almost linear increase in time against the test case size. This demonstrates to the better scalability of these LP based approaches in case of large-size optimization problems.

4.2. Case Study 2: Relative Optimality Comparison

In this study, TR-SLP is executed with the flat start strategy as well as with the DC warm start strategy for the OPF problem (2). As pointed out in [6, Section V], multi-start mode increases the probability of finding better local solutions for KNITRO-MS. Hence, we assume KNITRO-MS solutions as one of the best known solution sets for our benchmarking. Therefore in terms of

Table 2: The Relative Quality in the Optimality of TR-SLP Solution in Comparison to KNITRO-MS. TR-SLP is Executed under the Flat Start Strategy.

Test Case	Baseline Case			Thermally Constrained Case		
	Optimality ($/h)		η	Optimality ($/h)		η
	KNITRO-MS	TR-SLP		KNITRO-MS	TR-SLP	
IEEE 14	8,091	8,084	0.08%	9,294	9,278	0.17%
IEEE 30	575	578	−0.5%	582	584	−0.34%
IEEE 57	41,817	41,748	0.16%	41,978	42,003	−0.05%
IEEE 118	129,903	129,681	0.17%	135,189	135,103	0.06%
IEEE 300	720,149	722,314	−0.21%	726,794	728,981	−0.3%
Polish 2383	1,858,447	1,857,932	0.03%	1,863,627	1,862,370	0.07%
Polish 2746	1,185,115	1,208,279	−1.91%	1,185,507	1,208,278	−1.80%
Polish 3012	2,581,020	2,581,810	−0.03%	2,597,386	2,583,008	0.6%

Table 3: The Relative Quality in the Optimality of TR-SLP Solution in Comparison to KNITRO-MS. TR-SLP is Executed under the DC Warm Start Strategy.

Test Case	Baseline Case			Thermally Constrained Case		
	Optimality ($/h)		η	Optimality ($/h)		η
	KNITRO-MS	TR-SLP		KNITRO-MS	TR-SLP	
IEEE 14	8,091	8,088	0.03%	9,294	9,270	0.26%
IEEE 30	575	575	0%	582	583	−0.51%
IEEE 57	41,817	41,769	0.15%	41,978	41,961	0.04%
IEEE 118	129,903	129,895	0%	135,189	135,017	0.13%
IEEE 300	720,149	720,414	−0.03%	726,794	726,192	0.08%
Polish 2383	1,858,447	1,857,929	0.03%	1,863,627	1,862,377	0.07%
Polish 2746	1,185,115	1,208,277	−1.91%	1,185,507	1,208,281	−1.80%
Polish 3012	2,581,020	2,581,810	−0.03%	2,597,386	2,583,008	0.6%

optimality, we compare the quality of the solution obtained by TR-SLP against the KNITRO-MS solution. For Table 2 and Table 3, we define the following performance metric:

$$\eta = \frac{\text{KNITRO-MS} - \text{TR-SLP}}{\text{KNITRO-MS}} \times 100\% \tag{15}$$

where η indicates the relative improvement (if η is positive) or deterioration (if η is negative) in optimality of the solution obtained by TR-SLP with respect to the best known KNITRO-MS solution.

Table 2 reports the optimal solution obtained using KNITRO-MS and TR-SLP (with the flat start strategy) for the baseline and thermally constrained cases. In both cases, it can be observed that for half of the IEEE and Polish systems, TR-SLP results in slightly improved optimality with respect to the KNITRO-MS (ranging 0.03 % to 0.17 %). Performance of TR-SLP for the Polish 2746 system is observed to be slightly suboptimal for both the baseline and thermally constrained cases. However, KNITRO-MS is computationally slower than TR-SLP due to the multi-start feature, and represents a trade-off for using multi-start in practical OPF applications.

Table 3 reports the optimal solution obtained using KNITRO-MS and TR-SLP with DC warm start strategy for the baseline and thermally constrained cases. The convergence and optimality of the solution of TR-SLP depends upon the initial point. As such we noted TR-SLP with the DC warm start strategy locates slightly different optimal values compared to those of TR-SLP with the flat start strategy. Hence, in order to compare the relative effectiveness between the flat start and DC warm start strategies, we analyze the number of iterations taken under each strategy to converge to the final solution. Figure 2 depicts the TR-SLP iterations of the IEEE (118, 300) and Polish (2383, 2746, 3012) systems for both the baseline and thermally constrained cases. We observe that the DC warm start strategy speeds up the convergence compared to the flat start strategy, except for the Polish 2383 system. However, both strategies converge within the maximum number of iterations set for TR-SLP, *i.e.* $K = 30$.

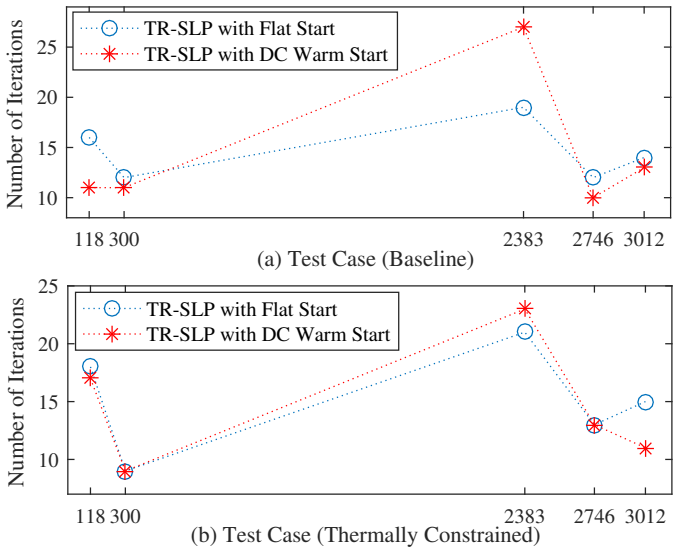


Figure 2: The relative comparison in iterations for the convergence of TR-SLP under flat start and DC warm start strategies for various IEEE (118, 300) and Polish (2383, 2746, 3012) systems.

4.3. Case Study 3: Discussion on Trust-Region Activation

In this study, we discuss the two main components in TR-SLP, *viz* the trust-region linear program (TR-LP) and feasibility restoration (NLP-FR). Specifically, we report the findings about the exact number of occurrences of TR-LP and NLP-FR in the execution of TR-SLP. We also study the effects of activation of a trust-region band on the execution of TR-SLP. The obtained results are reported for both the baseline and thermally constrained cases under the flat start strategy.

It can be seen in Figure 3 that NLP-FR is activated at least once in all the test cases, irrespective of the baseline or thermally constrained case. The NLP-FR was activated only once in all the test cases, except for the IEEE 118-bus system in the thermally constrained case, in which it was activated twice. Therefore, in all these cases the TR-SLP convergence is mainly governed by the TR-LP component. Furthermore, the relative propagation of the maximum

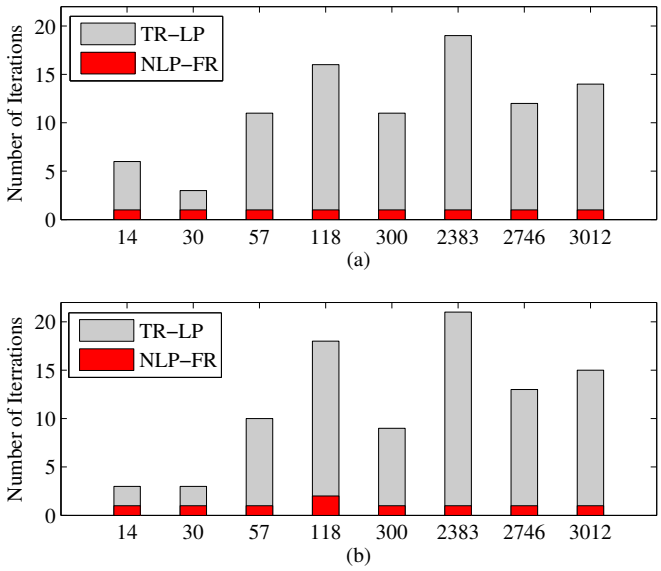


Figure 3: The relative comparison of NLP-FR and TR-LP executions for (a) baseline case and (b) thermally constrained case under flat start strategy for various IEEE (118, 300) and Polish (2383, 2746, 3012) systems.

step-size (d^k) along with a trust-region band ($[-\Delta^k, \Delta^k]$) is illustrated in Figure 4 for the Polish 2746 and 3012-bus systems. We note that NLP-FR has been first executed in the four simulations. Then, the trust-region initially expands due to insufficient closeness of the TR-LP to the original NLP. However, as the TR-SLP progresses the trust-region becomes smaller, converging to the maximum step-size solution within the specified solution tolerance band (ϵ_d). Overall, from Figure 4 it can be concluded that the trust-region assists convergence of the linear approximation neatly to the final solution.

4.4. Case Study 4: Relative Efficiency with Tightening the Tolerances

In this study, we further tighten the tolerances in TR-SLP and evaluate its impact on the quality of the OPF solution. For brevity, we restrict ourselves to the large test cases (Polish 2746, 3012) and TR-SLP is executed under the DC

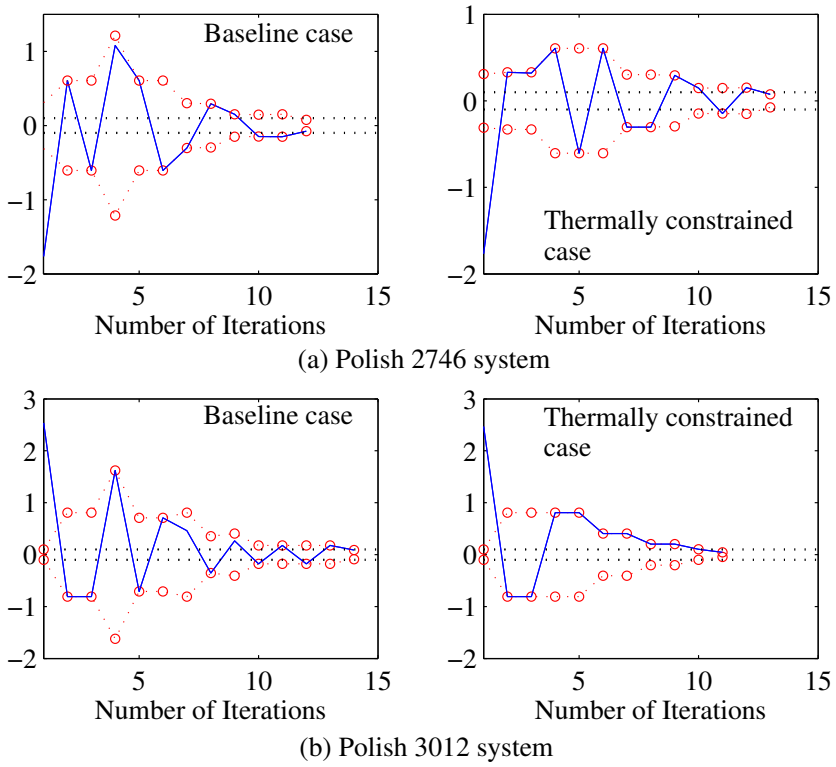


Figure 4: The relative propagation for the maximum step-size $\| d^k \|_\infty$ (solid line), trust-region band $[-\Delta^k, \Delta^k]$ (dotted $\circ$ line), and solution tolerance band $[-\epsilon_d, \epsilon_d]$ (dotted line) under flat start strategy for Polish (2746, 3012) systems.

warm start strategy. Initially, the tolerances are set to $\epsilon_d = 0.1,\ \epsilon = \epsilon_\lambda = 0.01$ (column A in Table 4), and are then tightened to $\epsilon_d = 0.01,\ \epsilon = \epsilon_\lambda = 0.001$ (column B in Table 4). The maximum number of iterations K is set to 100 in this case study. The results of the two experiments are showcased in Table 4. In comparison, the constraints are satisfied more accurately in the second experiment (as shown in column B) which is expected due to the tolerance tightening. However, we observed that this does not benefit the TR-SLP algorithm in a significant manner. For instance, the optimality improves only by very small amount relative to that of the first experiment (shown in column A); which is

less than 0.001% for Polish 2746 and approximately 0.0018% for Polish 3012 test systems. On the other hand, the total TR-SLP iterations and the corresponding computational time required to reach the desired optimality are increased by approximately 2 to 7 times. Therefore, this experiment shows that the original tolerances used in TR-SLP are good enough to reach the optimal solution while satisfying constraints for the test cases considered in the study.

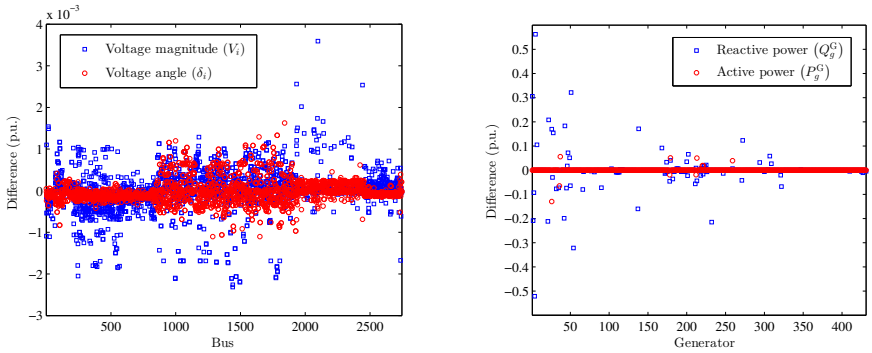
Table 4: Computational Results with Tightening the Tolerances $(\epsilon_d, \epsilon, \epsilon_\lambda)$ in TR-SLP under the DC Warm Start Strategy.

Test case	Performance metrics for TR-SLP		Tolerance settings A	Tolerance settings B
Polish 2746	Optimality ($/h)		1,208,281	1,208,279
	Iterations		13	89
	Computational time (s)		12.85	85.45
	Constraint satisfaction*	Max	2.07×10^{-4}	7.46×10^{-7}
		Mean	1.99×10^{-7}	1.09×10^{-9}
Polish 3012	Optimality ($/h)		2,583,008	2,582,962
	Iterations		15	35
	Computational time (s)		20.07	51.29
	Constraint satisfaction*	Max	7.59×10^{-5}	7.86×10^{-7}
		Mean	6.79×10^{-8}	5.45×10^{-10}

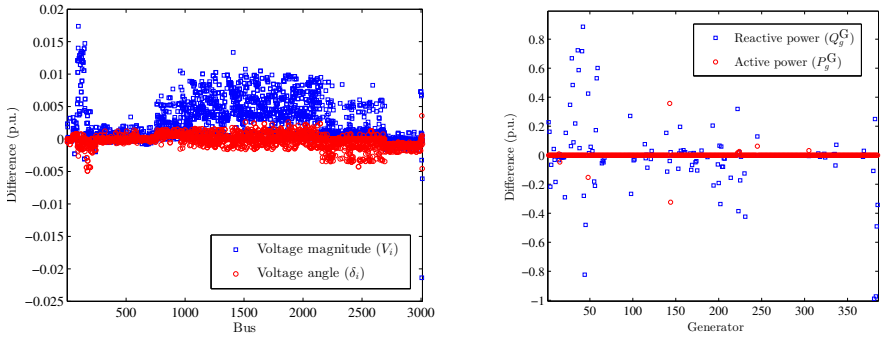
*indicates the accuracy up to which constraints (2b)−(2e) are satisfied.

In addition, let x_{A} be the solution of the experiment A and x_{B} be the solution of the experiment B. Figure 5 depicts the difference $(x_{\mathrm{A}} - x_{\mathrm{B}})$ in the final solutions of the two test systems in the two experiments. It can be observed that active power dispatch, which is related to the objective function, varies less than reactive power. Further, the voltage angle variation is also small. This may be due to the quasi-linear relationship of active power and voltage angles. However, the reactive power dispatch and voltage magnitudes are not included in the cost function, but only play a role in constraint satisfaction. Therefore,

we can observe slightly more variation in these two variables.



(a) Polish 2746 system.



(b) Polish 3012 system.

Figure 5: The difference between the bus voltages, voltage angles, active and reactive power values of the Polish 2746 system in (a) and Polish 3012 system in (b) under different tolerance settings (experiments A and B in Table 4). The vertical axis represents the magnitude difference $(x_A - x_B)$ in p.u.

5. Conclusions

We reported the classical AC-OPF formulation and proposed the SLP-based approach to solve it. The linear models formulated in the SLP approach are valid only at the vicinity of the linearization point. Therefore, a trust-region that bounds variations in the decision variables was introduced to tighten the SLP approximation. This ensures the convergence of SLP approximation for the

OPF problem (referred to as TR-SLP in this work). In addition, we also proposed the feasibility restoration phase based on the original nonlinear constraints to quickly locate a feasible point when the SLP approximation is infeasible. Results show that our TR-SLP approach outperforms KNITRO, IPOPT and a recently reported SLP method based on penalty reformulations [6] in terms of computational time.

We also used two *generic starting point strategies* (flat and DC warm start) and the OPF results on IEEE and Polish systems demonstrated the capability of TR-SLP to locate good local optimal solutions. It was observed, with these two starting strategies, in some cases, TR-SLP converges to a better solution than KNITRO-MS. It would be interesting to develop and study good starting point strategies for TR-SLP in future.

Acknowledgement

We acknowledge the support from the International Center of Energy Research (ICER), established by Nanyang Technological University, Singapore and Technische Universität München, Germany. We also acknowledge the support from the National Research Foundation (NRF), Prime Minister's Office, Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) programme.

References

[1] F. Capitanescu, J. M. Ramos, P. Panciatici, D. Kirschen, A. M. Marcolini, L. Platbrood, L. Wehenkel, State-of-the-art, challenges, and future trends in security constrained optimal power flow, Electric Power Systems Research 81 (8) (2011) 1731 – 1741.

[2] F. Capitanescu, Critical review of recent advances and further developments needed in ac optimal power flow, Electric Power Systems Research 136 (2016) 57 – 68.

[3] J. Lavaei, S. H. Low, Zero duality gap in optimal power flow problem, IEEE Transactions on Power Systems 27 (1) (2012) 92–107.

[4] W. A. Bukhsh, A. Grothey, . I. M. McKinnon, P. A. Trodde, Local solutions of the optimal power flow problem, IEEE Transactions on Power Systems 28 (4) (2013) 4780–4788.

[5] B. Stott, O. Alsaç, Optimal power flow−Basic requirements for real-life problems and their solutions, in: SEPOPE XII Symposium, Rio de Janeiro, Brazil, 2012.

[6] A. Castillo, P. Lipka, J-P. Watson, S. S. Oren, R. P. O'Neill, A successive linear programming approach to solving the IV-ACOPF, IEEE Transactions on Power Systems 31 (4) (2016) 2752–2763.

[7] S. Mhanna, G. Verbi, A. C. Chapman, Tight LP approximations for the optimal power flow problem, in: 2016 Power Systems Computation Conference (PSCC), 2016, pp. 1–7.

[8] J. Hörsch, H. Ronellenfitsch, D. Witthaut, T. Brown, Linear optimal power flow using cycle flows, Electric Power Systems Research 158 (2018) 126–135.

[9] A. Garces, A quadratic approximation for the optimal power flow in power distribution systems, Electric Power Systems Research 130 (2016) 222–229.

[10] D. Phan, J. Kalagnanam, Some efficient methods for solving the security-constrained optimal power flow problem, IEEE Transactions on Power Systems 29 (2) (2014) 863–872.

[11] F. Capitanescu, L. Wehenkel, Experiments with the interior-point method for solving large scale optimal power flow problems, Electric Power Systems Research 95 (2013) 276–283.

[12] V. H. Quintana, G. L. Torres, J. Medina-Palomo, Interior-point methods and their applications to power systems: a classification of publications

and software codes, IEEE Transactions on Power Systems 15 (1) (2000) 170–176.

[13] R. A. Jabr, A primal-dual interior-point method to solve the optimal power flow dispatching problem, Optimization and Engineering 4 (4) (2003) 309–336.

[14] S. Frank, I. Steponavice, S. Rebennack, Optimal power flow: a bibliographic survey I, Energy Systems 3 (3) (2012) 221–258.

[15] X. Bai, H. Wei, K. Fujisawa, Y. Wang, Semidefinite programming for optimal power flow problems, International Journal of Electrical Power and Energy Systems 30 (6-7) (2008) 383–392.

[16] H. Hijazi, C. Coffrin, P. V. Hentenryck, Polynomial SDP cuts for optimal power flow, in: 2016 Power Systems Computation Conference (PSCC), 2016, pp. 1–7.

[17] S. H. Low, Convex relaxation of optimal power flow-Part I: Formulations and equivalence, IEEE Transactions on Control of Network Systems 1 (1) (2014) 15–27.

[18] S. H. Low, Convex relaxation of optimal power flow-Part II: Exactness, IEEE Transactions on Control of Network Systems 1 (2) (2014) 177–189.

[19] R. Madani, S. Sojoudi, J. Lavaei, Convex relaxation for optimal power flow problem: Mesh networks, IEEE Transactions on Power Systems 30 (1) (2015) 199–211.

[20] B. Kocuk, S. S. Dey, Xu. A. Sun, Inexactness of SDP relaxation and valid inequalities for optimal power flow, IEEE Transactions on Power Systems 31 (1) (2016) 642–651.

[21] C. Coffrin, H. L. Hijazi, P. V. Hentenryck, The QC relaxation: A theoretical and computational study on optimal power flow, IEEE Transactions on Power Systems 31 (4) (2016) 3008–3018.

[22] M. Farivar, S. H. Low, Branch flow model: Relaxations and convexification−Part I, IEEE Transactions on Power Systems 28 (3) (2013) 2554–2564.

[23] M. Farivar, S. H. Low, Branch flow model: Relaxations and convexification−Part II, IEEE Transactions on Power Systems 28 (3) (2013) 2565–2572.

[24] H. Gao, J. Liu, L. Wang, Y. Liu, Cutting planes based relaxed optimal power flow in active distribution systems, Electric Power Systems Research 143 (2017) 272–280.

[25] Q. Zhijun, Y. Hou, Y. Chen, Convex envelopes of optimal power flow with branch flow model in rectangular form, IEEE Proceedings of Power and Energy Society General Meeting, Denver, CO (2015) 1–5.

[26] D. Li, X. Li, Decomposition-based global optimization for optimal design of power distribution systems, 55^{th} IEEE Proceedings of Control and Decision Conference, Las Vegas, USA (2016) 3265–3270.

[27] Z. Yang, H. Zhong, Q. Xia, A. Bose, C. Kang, Optimal power flow based on successive linear approximation of power flow equations, IET Generation, Transmission Distribution 10 (14) (2016) 3654–3662.

[28] A. Wächter, L. T. Biegler, On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming, Mathematical Programming 106 (1) (2006) 25–57.

[29] R. H. Byrd, J. Nocedal, R. A. Waltz, KNITRO: An integrated package for nonlinear optimization, Springer, 2006.

[30] S. Pajic, K. A. Clements, Power system state estimation via globally convergent methods, IEEE Transactions on Power Systems 20 (4) (2005) 1683–1689.

[31] W. Min, L. Shengsong, A trust region interior point algorithm for optimal power flow problems, International Journal of Electrical Power and Energy Systems 27 (4) (2005) 293–300.

[32] A. A. Sousa, G. L. Torres, C. A. Canizares, Robust optimal power flow solution using trust region and interior-point methods, IEEE Transactions on Power Systems 26 (2) (2011) 487–499.

[33] W. Sheng, K. Liu, S. Cheng, Optimal power flow algorithm and analysis in distribution system considering distributed generation, IET Generation, Transmission and Distribution 8 (2) (2014) 261–272.

[34] R. H. Byrd, Nicholas I.M. Gould, J. Nocedal, R. A. Waltz, An algorithm for nonlinear optimization using linear programming and equality constrained subproblems, Mathematical Programming 100 (1) (2004) 27–48.

[35] R. D. Zimmerman, C. E. Murillo-Sanchez, R. J. Thomas, MATPOWER: Steady-state operations, planning, and analysis tools for power systems research and education, IEEE Transactions on Power Systems 26 (1) (2011) 12–19.

[36] The IBM Corp., CPLEX Version 12.6.3, New York, US, 2016.